

Exame de Programação I

(Época de recurso, duração: 2 horas)

Universidade do Algarve

30 de Janeiro de 2002

- Não é permitido falar com os colegas durante a prova. Se o fizerem, terão a prova anulada.
- Não pode chamar o docente para lhe ajudar a interpretar o enunciado. A interpretação do enunciado faz parte da avaliação.
- A prova tem 5 perguntas e a cotação de cada uma aparece entre parêntesis.

Este exame tem um tema central—*permutações*. Uma permutação de n objectos, consiste em trocar a ordem desses objectos. Por exemplo, $3, 2, 5, 4, 1$ é uma permutação de $1, 5, 2, 4, 3$.

(1 valor) Pergunta 1. Faça uma função de nome *sequencia* que preencha um array com os números $1, 2, 3, \dots, n$.

```
void sequencia( int a[], int n )
{
    ...
}
```

(4 valores) Pergunta 2. Faça uma função que gere uma permutação aleatória dos números compreendidos entre 1 e n (nota: uma permutação aleatória de números é o mesmo que baralhar os números ao acaso). Para fazer este exercício, pode assumir a existência de uma função `random(a,b)` que gera um número inteiro aleatório no intervalo `[a..b]`.

DICA: o seguinte algoritmo faz aquilo que se pretende:

- (1): gerar uma sequência com os n primeiros números inteiros (1,2,3,...,n) e guardá-la num array 'a'.
- (2): `i = 0`
- (3): `k = número aleatório entre i e n-1`
- (4): trocar `a[k]` com `a[i]`
- (5): `i = i+1`
- (6): se `i < n-1` volta ao passo 3
- (7): termina

O passo 1 é pura e simplesmente a função feita na pergunta 1.

```
void permutacao( int p[], int n )
{
    ...
}
```

(3+3=6 valores) Pergunta 3. Um quadrado mágico é uma matriz de $n \times n$ na qual cada um dos inteiros $1, 2, 3, \dots, n^2$ está presente apenas uma vez e onde todas as somas de linhas, de colunas e das diagonais principais dão o mesmo valor. Por exemplo, este é um quadrado mágico de 5×5 no qual todas as somas referidas dão 65.

17	24	1	8	15
23	5	7	14	16
4	6	13	20	22
10	12	19	21	3
11	18	25	2	9

1. Faça um programa que permita guardar uma matriz (máxima dimensão 100×100). O programa deve pedir ao utilizador um número n e preencher as posições da matriz aleatoriamente com os números $1, 2, 3, \dots, n^2$. Para tal, pode (e deve) utilizar a função `permutacao` feita anteriormente.
2. Acrescente umas linhas de código à alínea anterior de modo a verificar se a matriz corresponde ou não a um quadrado mágico.

(1+2=3 valores) Pergunta 4. Considere a seguinte definição:

```
char s1[4][10] = {"lisboa", "porto", "faro", "coimbra"};  
char s2[4][10] = {"coimbra", "lisboa", "faro", "porto"};
```

1. Imagine que se pretendia detectar se s1 representa uma permutação de s2. Explique por palavras como é que isso poderia ser feito.
2. Transforme as palavras escritas na alínea anterior para um algoritmo (não é necessário programar, basta indicar um conjunto de passos, do mesmo modo que foi indicado na pergunta 2)

(6 valores) Pergunta 5. Escreva um programa que leia uma letra entre 'A' e 'Z' e produza uma pirâmide de letras. Por exemplo, se o input fosse a letra 'E', a pirâmide teria a seguinte forma:

```
  A  
 ABA  
ABCBA  
ABCDCBA  
ABCDEDCBA
```